SUPERVISED MACHINE LEARNING APPROACH FOR DETECTION OF MALICIOUS EXECUTABLES

YAHYE ABUKAR AHMED

A project submitted in partial fulfillment of the
requirements for the award of the degree of
Master of Computer Science (Information Security)

Faculty of Computer Science and Information Systems
Universiti Teknologi Malaysia

JANUARY 2013

This project is dedicated to my beloved brother for his endless support and encouragement, and to my parents.

ACKNOWLEDGEMENT

First and foremost, I would like to express heartfelt gratitude and my sincere appreciation to my supervisor **Professor Dr. Mohd Aizaini Maarof** and my co-supervisor **Dr. Anazida Zainal** for their constant support, encouragement, guidance and friendship. They inspired me greatly to work in this project. Their willingness to motivate me contributed tremendously to our project. I have learned a lot from them and I am fortunate to have their as my mentor and supervisor

Besides, I would like to thank my beloved brother who supported advices and financially, and also the authority of Universiti Teknologi Malaysia (UTM) for providing me with a good environment and facilities such as Computer laboratory to complete this project and software which I need during process.

ABSTRACT

Malware can be described as any type of malicious code that has the potential harm to the computer or network. these threats came from various sources like the internet, local networks and portable drives. Virus which replicates itself is growing faster every year and poses a serious global security threat. The purpose of this research is to classify portable executable new malicious files from benign files. In recent years, data mining methods are investigated for detecting unknown malicious executables, and the result show high and acceptable detection rate. Therefore, this project applied machine learning to detect malicious executable files through Support Vector Machine (SVM) and Artificial Neural Network (ANN) algorithms. These algorithms are compared together and selected the best accuracy model. The result of this research indicated that the accuracy of the SVM and ANN rely on the settings of the parameters used; ANN showed higher accuracy of 98.76 than SVM in terms of data set used while SVM performed a speed three times less than ANN and low computational power. The main conclusions drawn from this research were that current detection approaches of the antivirus are deficient because they fail to detect new unseen malicious files and they have higher false negative rates.

ABSTRAK

Malware boleh didefinisikan sebagai sebarang jenis kod pelbagai yang mempunyai potensi sebagai ancaman kepada komputer atau rangkaian dimana ancaman ini berpunca daripada pelbagai sumber seperti internet, rangkain setempat atau peranti luaran. Virus yang menyerupai diri sendiri berkembang dengan pesat setiap hari dan menyebabkan ancaman keselamatan di peringkat global. Tujuan kajian ini adalah untuk mengklasifikan pelaksanaan mudah alih pelbagai fail baru . Beberapa tahun kebelakangan ini, kaedah data mining telah disiasat untuk mengesan perlaksanaan kod pelbagai yang tidak diketahui puncanya dan keputusan menunjukkan kadar pengesanan yang tinggi dan diterimapakai. Dengan itu projek ini akan menggunapakai pembelajaran mesin untuk mengesan pelaksanaan pelbagai fail melalui algoritma Sokongan Mesin Vektor (SVM) dan Rangkaian Neural Kebijaksanaan (ANN) . Algoritma ini akan dibandingkan dan model yang paling tepat akan dipilih, Keputusan kajian akan menunjukkan ketepatan SVM dan ANN bergantung ke atas konfigurasi parameter yang digunakan dan kajian menunjukkan ANN mempunyai tahap ketepatan 98.76 berbanding SVM daripada segi konfigurasi parameter yang digunakan dengan kelajuan prestasi SVM adalah kurang tiga kali berbanding ANN dan kuasa pengkomputeran juga adalah rendah. Kesimpulan yang dapat dibuat daripada kajian ini adalah pendekatan pengesanan antivirus sedia ada masih banyak kekurangan kerana gagak untuk mengesan fail mengandungi kod pelbagai yang tidak dapat dilihat dan mereka mempunyai kadar negatif penipuan yang lebih tinggi.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Introduction

The most popular computer world attack is malware that presented a serious threat to the business and computer home users. Recent statistics from Microsoft Windows Malicious Software Removal Tool (MSRT) shows that about 5.7 million of specific Window computers users over the last one and half years are infected by one or more malicious codes (Jianyong *et al*, 2011).

Malware are programs that infect a computer system without the user's known permission. It performs malicious actions on the victim machine such as collecting of information that causes a loss of personal information or misuse properties, and accessing unauthorized resources of the system (Priyank and Nataasha, 2012). Therefore, Malware comes in a variations and different range of forms and such as computer viruses, Trojan horses, spyware, worms, adware, rootkits, and other crime or abuses related to the computer (Priyank and Nataasha, 2012).

The most common detection method used by the antivirus is the signature based detection; it is based on the characterization of the knowledge in its repository. The signature technique is becoming more difficult and detectable, since all recent malware applications have intention of getting a number of stealth techniques to evade detection. Some of malware writers use obfuscation technique to automatically

modify themselves to a unknown version at short range of periods in order to evade antivirus software detection (Muazzam *et al.*, 2005; Dragos *et al.*, 2009). The consequence of this problem make signature based detection less efficient and reliable.

Machine learning (ML), a broad branch of artificial intelligence, is computational methods using regularities, induced patterns and previous experience to improve accurate predictions and performance. It is designed to develop the efficiency of computer algorithms to solve a large-scale of data. In machine learning classifier is used to recognize contents inside executable code files to classify new files from normal files (Eitan, 2009). Classifier is a set o f rules that is applied to a specified training of malicious executables and normal files.

Generally, classifiers are trained to recognize unseen malicious executables as maliciousness, and complex patterns recognition that lead intelligent decisions based on the training data. In machine learning algorithms track the sequences that generated by the system calls and addressed as the characteristics of the program. The programs interact with the operating system through system calls. Therefore, the input of machine learning algorithms depend on the feature selection including API calls.

The remaining of this chapter is organized as follows; the first section will discuss on problem background. Problem statement is detailed in the second section. The rest are scope, the objectives of the project and significance of the Study.

1.2 Problem Background

Detection of known malware is commonly performed by tools such as anti-virus programs based on signature recognition. When a new malicious piece is discovered, the anti-virus vendors need to catch its binary signature through analysis of the instructions that make up the executable code. Virus detection program searches the virus signature database to find if there are matched signatures. If a match is found, the file under test will be identified as a malicious executable (Eitan, 2009). The main requirement of the system is to have an updated database of all the signature files of malware.

This approach has proved to be effective when the malware are known beforehand in the database, and the accuracy is totally dependent on the signature database of the system. However, Signature based detection systems have several issues made it less than completely reliable: it cannot detect a new unseen malware and easily escape the detection by simple defenses like code obfuscation, since the database will not have any information about the previously new virus (Yoshiro *et al.*, 2010).

During the time between the emergence of the unseen malicious files and the antivirus updating of the database released, millions of home computers users and network systems are vulnerable to the new threat. Using this method is deficient against unknown malware (Christodorescu and Jha, 2004; White, 1999). The second technique involves heuristic-based methods based on expert's rules to determine the behavior of malicious files, or normal behavior. Although it is reliable and capable of detecting new viruses, but this method is no longer considered effective and efficient compared against the high spreading rate of malware, because the chances of false alarm are relatively higher (Jacob *et al.*, 2008; Gryaznov, 1999).

In recent years, Machine learning approaches are investigated for detecting unknown viruses, and the result show high and acceptable detection rate. Machine learning techniques can also detect metamorphic viruses since they can be used to detect patterns between the generations of a specific family of virus. In machine learning, (Kolter, Maloof, 2006) proposed a text classification method that detected and indentified malicious binary executable file. They compare machine learning classifiers such as, Naive Bayesian, Decision Tree, boosted decision tree and Support Vector Machine (SVM), the result showed that boosted decision tree had a perfect performance.

Many researches concerned about the detection of executable binary files using machine learning algorithms to test specific method. In (Jingbo *et al.*, 2010; Bo-yun, 2006) presented a method for detecting previously unseen polymorphic computer viruses. Their model was based on SVM algorithm using system API to detect malicious codes. Although the sample dataset size was small, they showed the model's detection accuracy and performance better by comparing the result of SVM with other learning algorithms.

In research Yingxu and Zhenghui (2011) proposed a new improved feature selection method which can represent the most properties. This method only extract features from benign programs, and build a one-class classifier. When identifying the testing data with the classifier, they will be classified as either malicious or benign. This method achieved better performance and showed high Performance Detection Rate (PDR); it can identify the previously unseen malicious executables from normal files.

In recent year Singhal and Raul (2012) have proposed advanced data mining and machine learning for malware detection. Their module extracts API calls made by various normal and harmful executable. Their proposed method implemented enterprise gateway level to act as a supplement anti-viruses present on end user computers.

Two main important characteristic of the performance and accuracy of machine learning classifier algorithms are: the first is the inducer algorithm employed to produce the algorithm, second is the type of features extracted from files. In ZicoKolter and Maloof (2006) have proposed method to detect malicious executables in wild. They have used N grams of both malicious and benign programs; their data set contains 1971 benign files and 1651 malicious executables files as features. Although, they showed high performance based on the accuracy of classifying dirt files from normal file(Silvio, 2010), but N grams is not longer effective to use as feature for detecting polymorphic and metamorphic malware, because of its sensitivity to order.

In the neural networks community ensemble has been proposed by several authors (Boyun, 2007; GangLiu et al., 2010; Muhammad et al., 2011). Their method is based on multi-classifier combination using Dempster-Shafer theory as evidence for detecting unseen malicious code. This method has improved the detection accuracy of Portable Executable (PE) based malware on Windows platforms. They used API calling sequences as feature extraction from suspicious files and different types of learning machine to construct the ensemble. Their result indicated higher accuracy and performance over individual classifier output. However, ensemble method has problem of overfitting which occurs when a model has many parameters that causes excessively complexity.

Performance of supervised machine learning algorithms is influenced by two different issues: the representational feature and type of model used, therefore, the extraction of the API call function sequence as feature in static analysis method with a controlled environment could lead to monitor and trace the malicious behavior of files. This research focused on API calls function as feature representation method.

1.3 Problem Statement

The traditional detection accuracy (signature based) of malware is ineffective, because of constantly changing of malware nature and shapes through obfuscation techniques. Feature representation of PE can increase the accuracy of malware detection in data mining techniques. Some feature representations are no longer effective to be input of supervised machine learning algorithms while others showed higher performance detection rate; there are important questions which arise:

1. Feature extraction is a key to apply machine learning to successfully detect malicious executables, which feature extraction approach can propose significant features that can be represent malware?
2. How supervised machine learning can be implemented using API calls as feature extraction?
3. How to apply Neural Network and Support Vector Machine model to detect malicious executables?
4. How to measure the efficiency of Neural Network and Support Vector Machine classifiers in detecting of malware.

1.4 Project Aim

The purpose of this project is to examine the machine learning techniques in the domain of classifying benign and maliciousness executables programs, and form a comparison of their effectiveness and accuracy through cross-validation test in a controlled environment.

1.5 Objectives of the Project

The following are the objectives of the project:

1. To perform data pre-processing that will prepare appropriate format to be input to Machine Learning classifiers.
2. To develop two representatives supervised machine learning models; Support vector machine and artificial neural network.
3. To evaluate the performance of Support vector machine and artificial neural network to classify for new malicious executables programs.

1.6 Scope of the Project

In this project, machine learning algorithms will be used for classification of dataset as benign or malicious executables. Therefore, the scope of this project will be the following:

1. Focus on malicious program that exists in Microsoft Windows as experiment platform and VMware as virtual machine. Specially, portable executable files because Windows has such a large share of the personal computing market, in addition to malicious writers target the Microsoft Window.
2. Inputs will be restricted to a single feature type: Application program interface (API) calls. Because API calls can expose implicit features of the input that are complicated to detect explicitly.
3. In this project, Supervised Machine learning techniques will be focused, because it performs statistical comparisons on specific datasets to examine the accuracies of trained classifiers.
4. Four common performance metrics will be used, so evaluate the performance of ensemble machine learning technique are True Positive (TP), False Positive (FP), True Negative(TN), and finally False Negative (FN).

1.7 Significance of the Study

The expected outcome of this study is to detect malicious executable files with better accuracy in comparison to signature based detection methods. This proposed scheme can be used in real-life situations such as computer users and organization network. Efficient malware detection can save financial loss and provide computer home user confidence in the security field. Supervised machine learning approach is expected to have higher performance while maintaining low false positives. This study will also be beneficial to the ant-virus researches through effective machine learning algorithms, and provide recommendations on how to evaluate the performance of a certain ML algorithms in accordance to malware detection.

1.8 Organization of the Report

This study consists of six chapters. Chapter 1 is about introduction of study, Problem background, objectives, scope and significance of the project. Chapter 2 provides the literature reviews on malware detection methods including machine learning algorithms. The framework of methodology and data set used to detect new executable malicious files will be discussed in the Chapter 3. Chapter 4 data preprocessing steps, feature extraction and feature selection. Developing models, evaluating and comparing of the proposed method based on the accuracy of the algorithms are discussed in Chapter 5. Finally, in Chapter 6, the conclusion, research contribution and further work to be conducted in this research are discussed.

REFERENCES

Abhishek singh, Baibhav Singh and Hirosh Joseph (2008). Malware analysis, Advances in Information Security ,Volume 37, 2008, DOI: 10.1007/978-0-387-74390-5

Aditya Govindaraju. (2010). Exhaustive Statistical Analysis for Detection of Metamorphic Malware, Master's Projects, San Jose State University

Arini Balakrishnan and Chloe Schulze.(2005). Code obfuscation literature survey. Computer Sciences Department, University of Wisconsin, Madison

Asaf Shabtai, Robert Mosk ovitch,Yuval Elovici, Chan an Gle zer. (2009). Detection of malicious code by applying machine learning classifier on static features: A state-of-the-art survey, Ben-Gurion University, Beer Sheva, Israel.

Asaf Shabtai, Robert Moskovitch, Clint Feher, Shlomi Dolev and Yuval Elo vici.(2010) . Detecting unknown malicious code by applying classification techniques on OpCode patterns Ben-Gurion University, Be'er Sheva, 84105, Israel.

Ben Kröse & Patrick van der Smagt, (1996). An introduction to neural networks, available at URL:http://www.fwi.uva.nl/research/neuro/

Boyun Zhang, Jianping Yin, Jingbo Hao, Dingxing Zhang, Shulin Wang .(2007). Malicious codes detection based on ensemble learning, School of Computer Science, National University of Defense Technology, Changsha, China.

Boyun Zhang, Jianping Yin, Jingbo Hao, Dingxing Zhang, Shulin Wang .(2007). Using Support Vector Machine to Detect Unknown Computer Viruses, School of Computer Science, National University of Defense Technology, Changsha, China.

Christodorescu and S. Jha M. (2003). Static analysis of executables to detect malicious patterns, In Proceedings of the Usenix Security.

Craig Steven Wright. (2010). Debugging and unpacking the NsPack 3.4 and 3.7 packer, SANS Institute InfoSec Reading Room.

David Hecherman. (1995). A tutorial on learning bayesian networks, technical Report, Microsoft Research Advanced Technology Division, Microsoft Corporation

Dragos, Gavrilut, Mihai Cimpoes, Dan Anton,Liviu Ciortuz.(2009). Malware detection using machine learning. University of Iasi, Romania.

Eitan Menahem, Asaf Shabtai, LiorRokach,YuvalElovici. (2009). Improving malware detection by applying multi-inducer ensemble, Ben-Gurion University of the Negev, BeerSheva, 84105, Israel.

Evgenios Konstantinou and Stephen Wolthusen.(2008). Metamorphic virus: analysis and detection, white paper, Royal Holloway, University of London.

Fernando Fernandez-Rodrıguez, Christian Gonzalez-Martel and Simon Sosvilla-River .(2000).“On the profitability of technical trading rules based on artificial neural networks”, Universidad de Las Palmas de Gran Canaria, Las Palmas , Canary Islands, Spain.

Gang Liu, Wei Chen. (2012). A neural network ensemble based method for detecting computer virus, College of Computer Science and Engineering Chang chun University of Technology Chang chun, China.

Gentleman R., W. Huber, and V. J. Carey (2008). Supervised machine learning http://www.spamlaws.com/how-worm-malware-works.html

IBM Corporation. (1996). Neural networks for computer virus recognition, Retrieved April 10, 2012, http://www.research.ibm.com/antivirus/SciPapers/Tesauro

Igor Santos, Carlos Laorden and Pablo G. Bringas.(2003). Collective classification for unknown malware detection, University of Deusto Avenida de las University 24, 48007, Bilbao, Spain,

Ilsun You and Kangbin Yim.(2010). Malware obfuscation techniques: A Brief survey international conference on broadband, Wireless Computing, Communication and Applications

Jianyong Dai, Ratan Guha, Joohan Lee. (2011) Efficient virus detection using Dynamic instruction sequences, University of Central Florida, Orlando, Florida.

Jingbo Sun, Wei Chen, Fen Hu. (2010). A SVM-based method for detecting computer virus, Changchun University of Technology Changchun, China.

Kohavi and R. Quinlan. (1999). Decision tree discovery, Handbook of data mining and knowledge discovery, W. Klosgen and J.M. Zytkow , eds., New York, NY: Oxford University Press, pp. 267-276.

Lars Arne Sand (2011). Malware detection: A review of the state of the art malware detection techniques IMT 4022 digital forensics ,Gjøvik University college Gjøvik, Norway.

Liu Wu, Ren Ping, Liu Ke and Duan Hai-xin.(2011). Behavior-based malware analysis and detection, first International Workshop on Complexity and Data Mining. Network Research Center of Tsinghua University, 100084 Beijing, P. R. China

Mamoun Alazab, Sitalakshmi Venkataraman and Paul Watters. (2010). Towards understanding malware behavior by the extraction of API calls, Second Cybercrime and Trust worthy Computing Workshop, University of Ballarat.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard fahringer Peter Reuteman and Ian H.Witten.(2008). The WEKA data mining software: an update, University of Waikato, Hamilton, New Zealand.

Menahem E. Troika.(2008). An improved stacking schema for classification tasks, M.Sc. thesis. Ben-Gurion University of the Negev. Israel

Miao Wang, Cheng Zhang and Jingjing Yu. (2006). Native API based windows anomaly intrusion detection method using SVM, School of Electronic and Information Engineering, Xi'an Jiaotong University, China.

Michael Sikorski and Andrew Honig.(2012).Practical malware analysis, William Pollock, San Francisco.

Microsoft Corporation. (1999). Microsoft portable executable and common object file format specification, Microsoft Corporation, 6 editions,

Mila Dalla Preda.(2007). Code obfuscation and malware detection by abstract interpretation, Ph.D Thesis. Universita degli Studi di Verona

Mitchell T. (1997). Machine learning, McGraw-Hill.

Muazzam Ahmed Siddiqui.(2008). Data mining methods for malware detection, College of Sciences at the University of Central FloridaOrlando, Florida

Muazzam Siddiqui, Morgan C. Wang, Joohan Lee. (2005). Detecting internet worms using data mining techniques. University of Central Florida.

Muhammad Najmi Ahmad Zabidi, Mohd Aizaini Maarof, Anazida Zainal. (2011). Ensemble based categorization and adaptive model for malware detection, 978-1-4577-2155- IEEE

Oded Maimon and Lior Rokach .(2005). Introduction to supervised methods, Book chapter 6, Tel-Aviv University.

Péter Ször. (2000). Attacks on Win32–Part II, Virus, vol. 47,

Priyank Singhal and Nataasha Raul. (2012).Malware detection module using machine learning algorithms to assist in centralized security in enterprise networks, International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.1, University of Mumbai, Mumbai, India

Quanquan Gu, Zhenhui Li and Jiawei Han.(2010). Generalized fisher score for feature Selection, University of Illinois at Urbana-Champaign, Urbana, IL 61801, US

Rishna Sandeep Reddy D K and Arun K Pujari. (2006). N -gram analysis for computer virus detection, University of Hyderabad, Hyderabad 500 046, Indi

Srilatha Attaluri.(2007). Detecting metamorphic viruses using profile hidden markov models, Master's Projects, Department of Computer Science San Jose State University

Sujandharan Venkatachalam. (2010). Detecting undetectable computer viruses. Master's theses. San Jose State University,

Tan P.-N., M. Steinbach, and V. Kumar.(2004). Introduction to Data Mining . Book chapter, MA: Addison-Wesley.

Thomas E. Dube, BCE, Captain and Usaf. (2006). Metamorphism as a software protection for non-malicious code, Master thesis, Air Force Institute of Technology Air University

Thomas G. Dietterich.(2000). Ensemble methods in machine learning, Oregon State University, Corvallis, Oregon, USA.

Ulrich Bayer, Andreas Moser,Christopher Kruegel and Engin Kirda.(2006). Dynamic analysis of malicious code, Technical University Vienna,Vienna, Austria

Vikramaditya Jakkula.(2007). Tutorial on Support Vector Machine, School of EECS, Washington State University,Pullman.

Vinod P, V.Laxmi,M.S.Gaur. (2009). Survey on malware detection methods, Malaviya National Institute of Technology, Jaipur, Rajasthan

Visual C++ Business Unit (1994). Microsoft portable executable and common object File format specification 4.1, MSDN Library, Microsoft Corporation.

Whil Hentzen.(2003). How viruses, worms, and Trojans work, Book Chapter. www.hentzenwerke.com

Wing Wong and Mark Stamp.(2006). Hunting for metamorphic engines

Yingxu Laia, Zhenghui Liu. (2011). Unknown malicious code detection based on bayesian, College of Computer Science, Beijing University of Technology, Beijing,China.

Yoshiro Fukushima, Akihiro Sakai, Yoshiaki Hori and Kouichi Sakurai(2010). A behavior based malware detection scheme for avoiding false positive, Graduate School of Information Science and Electrical Engineering, Kyushu University, Motoka, Nishi-ku, Fukuoka, Japan.

ZicoKolter KOLTER and Marcus A.Maloof.(2006).). Learning to detect and classify malicious executables in the Wild, Stanford University Stanford, CA94305-9025, USA